MULTIMEDIA UNIVERSITY

STUDENT ID NO

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 1, 2018/2019

TCP2451 – PROGRAMMING LANGUAGE TRANSLATION / TCS3311 – COMPILER DESIGN

(All sections / Groups)

26 OCTOBER 2018
3:00 p.m. – 5:00 p.m.
(2 Hours)

INSTRUCTIONS TO STUDENTS

1. This Question paper consists of **5 pages** only including the cover page with 4 Questions.

2. Attempt **ALL** questions. All questions carry equal marks and the distribution of the marks for each question is given.

3. Please print all your answers **CLEARLY** in the Answer Booklet provided.

Question 1 (6+6.5 marks)

(a) Illustrate six phases of a compiler using a diagram. Briefly explain each phase.

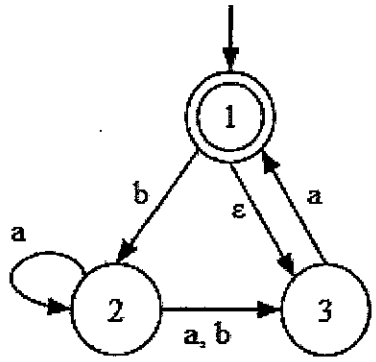
(b) Answer true or false for each of the following statements.

i. Regular expressions cannot check balancing tokens such as brackets.
ii. Context free grammar is a helpful tool in describing the syntax of programming languages.
iii. Optimization can be assumed as something that removes unnecessary code lines and arranges the sequence of statements in order to speed up the program execution without wasting resources such CPU, memory.
iv. There are only two parts of compilation which are analysis and synthesis.
v. All programming language constructs can be described by context-free grammars.
vi. Parsers do not parse the whole code if some errors exist in the program.
vii. Java is not a hybrid translator.

Continued.....

Question 2 (9+3.5 marks)

(a) Consider the following NFA.



i. Explain what ε-closure of a state is using an example from this NFA.
ii. Convert the NFA to a DFA using subset construction algorithm. Provide all the steps, the transition table, and the DFA.

(b) Consider the following grammar.

$E \rightarrow T\ X$
$T \rightarrow (\ E\ ) \mid \text{int}\ Y$
$X \rightarrow +\ E \mid \varepsilon$
$Y \rightarrow *\ T \mid \varepsilon$

i. Find the set for the FOLLOW(E).
ii. Find the set for the FOLLOW(T).
iii. Find the set for the FOLLOW(X).
iv. Find the set for the FOLLOW(Y).

Continued.....

Question 3 (4+3+4+1.5 marks)

Given the following Java CUP specification file.

```
package myparser;

import java_cup.runtime.lr_parser;
import java_cup.runtime.Scanner;
import java_cup.runtime.Symbol;
import java_cup.runtime.virtual_parse_stack;

/* REMOVED */

/* The grammar */
program ::= header body;

header ::= STUDENTID COLON STUDENTNAME;

body ::= LBRACE sentence RBRACE;

sentence ::= subject verb object PERIOD
             {: System.out.println("This is a sentence."); :}
             ;

subject ::= I | THE noun;

object ::= ME | THE noun;

noun ::= CATS | BATS | RATS;

verb ::= LIKE | SEE;
```

(a) Complete the "REMOVED" commented section of this file so that it can be used to generate a parser.

(b) Give an example of a source program that conforms to this syntax. You may specify any logical assumptions on the symbols used.

(c) Modify the grammar above so that the program body can contain one or more statements. Explain with an example how the changes can allow more statements.

(d) Write THREE regular expression rules for the JLex specification file so that the scanner generated can be used together with the Java CUP parser generated from this file.

Continued.....

Question 4 (0.5+5+4+3 marks)

(a) State one out of two semantic type checkings.

(b) Convert the expression (a + -(b+c)) into each of the following.
   i. syntax tree
   ii. three-address code
   iii. quadruples
   iv. triples
   v. indirect triples

(c) Construct both the unoptimized syntax tree and directed acyclic graph (DAG) respectively for the expression ((x + y)*(x – y)).

(d) Consider the following machine instruction sequence.

```
MOV y, R0
ADD z, R0
MOV R0, x
```

Convert the machine instruction sequence to a Java statement.

End of Page